

## 使用布局编辑器构建界面

在布局编辑器中，您可以通过将界面元素拖动到可视化设计编辑器中（而不是手动编写布局 XML），快速构建布局。设计编辑器支持在不同的 Android 设备和版本上预览布局，并且您可以动态调整布局大小，以确保它能够很好地适应不同的屏幕尺寸。

使用 `ConstraintLayout`（在支持库中提供的布局管理器，与 Android 2.3（API 级别 9）及更高版本兼容）构建新布局时，布局编辑器的功能尤其强大。

本文简要介绍了布局编辑器。要详细了解布局基础知识，请参阅[布局](#)。要详细了解如何使用 `ConstraintLayout` 构建布局，请参阅[使用 ConstraintLayout 构建自适应界面](#)。

### 布局编辑器简介

当您打开 XML 布局文件时，就会出现 Layout Editor。

编辑器的各个区域（对应于图 1 中的各个数字）如下：

1. **Palette**：您可以拖到布局中的视图和视图组列表。
2. **Component Tree**：查看布局的层次结构。
3. **Toolbar**：用于在编辑器中配置布局外观及更改某些布局属性的按钮。
4. **Design editor**：Design 视图和/或 Blueprint 视图中的布局。
5. **Attributes**：用于对所选视图的属性进行控制的控件。

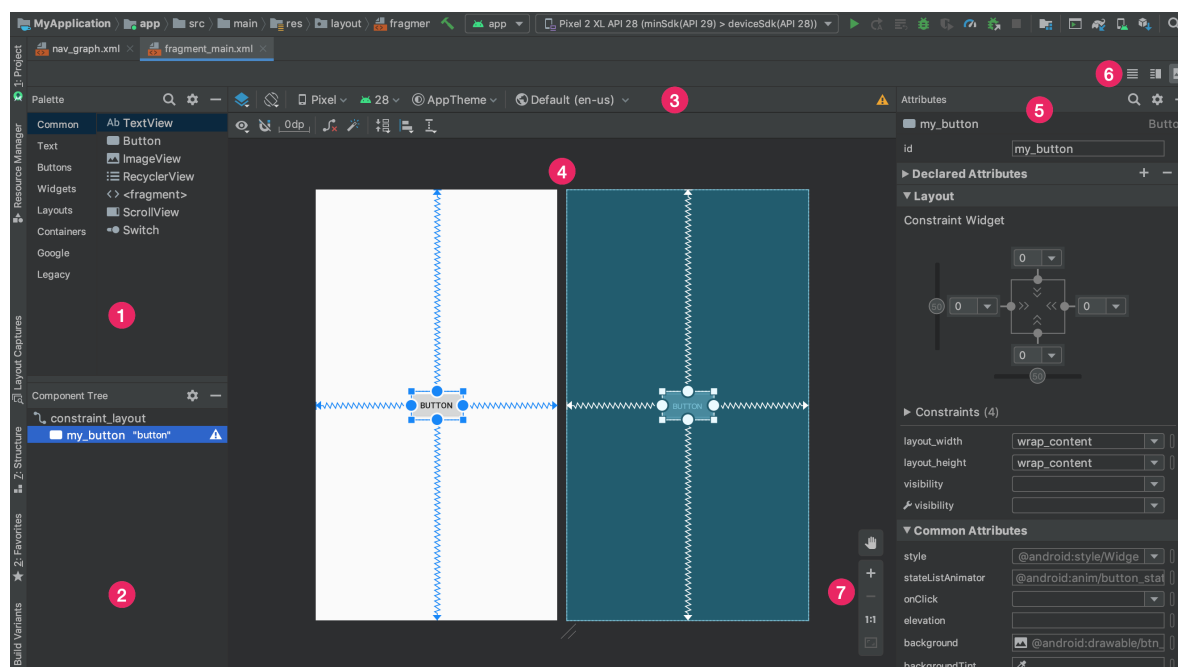


图 1. 布局编辑器

当您打开 XML 布局文件时，默认会打开设计编辑器（如图 1 所示）。

要在文本编辑器中修改布局 XML，请点击窗口底部的 **Text** 标签。在文本编辑器中，您还可以通过点击窗口右侧的 **Preview**，查看 **Palette**、**Component Tree** 和设计编辑器。文本编辑器中没有 **Attributes** 窗口。

**提示：**您只需按 **Alt + Shift + 向右/向左箭头**（在 Mac 上按 **Control + Shift + 向右/向左箭头**），即可在设计编辑器和文本编辑器之间切换。

### 更改预览外观

您可以使用设计编辑器顶行中的按钮在编辑器中配置布局的外观。文本编辑器的 **Preview** 窗口中也有此工具栏。

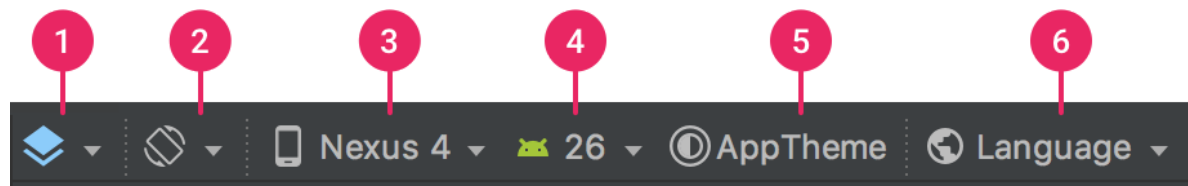


图 2. 布局编辑器工具栏中用于配置布局外观的按钮

可用的按钮（对应于图 2 中的各个数字）如下：

1. Design and blueprint

：选择您希望如何在编辑器中查看布局；可以选择

Design

视图（布局的实际预览效果）、

Blueprint

视图（仅显示每个视图的轮廓）或

Design + Blueprint

（并排显示这两个视图）。

**提示：**按 B 可循环浏览这些视图。

2. **Screen orientation and layout variants**：选择屏幕方向（横向和纵向），或应用提供备用布局的其他屏幕模式（例如夜间模式）。该菜单还包含用于[创建新布局变体](#)的命令。

3. Device type and size

：选择设备类型（手机/平板电脑、Android TV 或 Wear OS）和屏幕配置（尺寸和密度）。您可以从多种预配置的设备类型和您自己的 AVD 定义中进行选择，也可以从列表中选择

Add Device Definition

来启动新的 AVD。

**提示：**您可以通过拖动布局的右下角来调整设备尺寸。

4. **API version**：选择预览布局时使用的 Android 版本。

5. **App theme**：选择要应用于预览的界面主题背景。（这仅适用于支持的布局样式；因此该列表中的许多主题背景都会导致出错。）

6. **Language**：选择要以何种语言显示界面字符串。此列表仅会显示您的字符串资源支持的语言。如果您想要修改翻译，请点击下拉菜单中的 **Edit Translations**（请参阅[使用 Translations Editor 本地化界面](#)）。

**注意：**这些配置对应用的代码或清单没有任何影响（除非您选择从 **Layout Variants** 添加新的布局文件）；它们只会影响布局预览。

## 创建新布局

为应用添加新布局时，请先在项目的默认 `layout/` 目录中创建一个布局文件，以便其适用于所有设备配置。创建默认布局之后，您可以为特定的设备配置（例如特大屏幕）[创建布局变体](#)。

您可以通过几种不同的方式创建新布局（具体取决于您的 **Project** 窗口视图），但可以在任意视图中按以下步骤操作：



1. 在 **Project** 窗口中，点击要在其中添加布局的模块（如 **app**）。
2. 在主菜单中，依次选择 **File > New > XML > Layout XML File**。
3. 在随即显示的对话框中，输入文件的名称、根布局标记以及布局所属的源代码文件集，然后点击 **Finish**。

您还可以通过以下几种方式启动新布局文件（但显示的对话框会有所不同）：

- 如果您在 **Project** 窗口中选择了 **Project** 视图：打开应用模块的 **res** 目录，右键点击您要添加布局的布局目录，然后依次点击 **New > Layout resource file**。
- 如果您在 **Project** 窗口中选择了 **Android** 视图：右键点击 **layout** 文件夹，然后依次选择 **New > Layout resource file**。

## 创建布局变体

如果您已经有布局，并且想创建备用版本以针对不同屏幕尺寸或方向优化布局，请按照以下步骤操作：

1. 打开原始布局文件，并确保您正在查看的是设计编辑器（点击窗口底部的 **Design** 标签）。
2. 点击工具栏中的 **Orientation for Preview** 图标 。在下拉列表中，点击建议的变体（例如 **Create Landscape Variant**），就大功告成了；或点击 **Create Other**，然后继续执行下一步。
3. 在随即显示的对话框中，您只需定义目录名称的资源限定符即可。您可以在 **Directory name** 中输入限定符，也可以从 **Available qualifiers** 列表中选择（一次选择一个），然后点击 **Add** 图标 。
4. 添加完所有限定符后，点击 **OK**。

如果同一布局有多个变体，您可以从点击 **Layout Variants** 图标  后显示的列表中轻松切换它们。

如需详细了解如何为不同屏幕创建布局，请参阅[支持不同的屏幕尺寸](#)。

## 转换视图或布局

您可以将一种视图转换为另一种视图，也可以将一种布局（视图组）转换为另一种布局。

1. 点击编辑器窗口底部的 **Design** 标签。
2. 在 **Component Tree** 中，右键点击相应视图或布局，然后点击 **Convert view**。
3. 在随即显示的对话框中，选择新类型的视图或布局，然后点击 **Apply**。

## 将布局转换为 ConstraintLayout

`ConstraintLayout` 是 Constraint Layout 库（Android Studio 2.2 及更高版本中包含该库）中提供的一个视图组。它是与布局编辑器一起从头开始构建的，因此您可以从设计编辑器中访问其中的所有内容，而且您永远无需手动修改 XML。最重要的是，它采用基于约束条件的布局系统，让您无需任何嵌套视图组即可构建大多数布局。

为了改善布局性能，您应该将旧版布局转换为 `ConstraintLayout`。

要将现有布局转换为 `ConstraintLayout`，请执行以下操作：

1. 在 Android Studio 中打开现有布局，然后点击编辑器窗口底部的 **Design** 标签页。

## 2. 在 **Component Tree** 窗口中，右键点击该布局，然后点击 **Convert layout to ConstraintLayout**。

与上一部分中介绍的简单 **Convert view** 命令相比，专门用于将布局转换为 `ConstraintLayout` 的命令在推断约束条件和保留布局方面更加智能。

要详细了解如何使用 `ConstraintLayout` 构建布局，请参阅[使用 ConstraintLayout 构建自适应界面](#)。

在 Palette 中查找内容

要在 **Palette** 中按名称搜索视图或视图组，请点击 Palette 顶部的 **Search**  按钮，或者在 **Palette** 窗口处于活跃状态时直接开始输入内容名称。

您可以在 **Palette** 内的 **Common** 类别中找到经常使用的内容。要向此类别中添加内容，请右键点击 **Palette** 中的相应视图或视图组，然后在上下文菜单中点击 **Favorite**。

从 Palette 中打开文档

要打开视图或视图组的 Android 开发者参考文档，请在 **Palette** 中选择界面元素，然后按 Shift+F1。

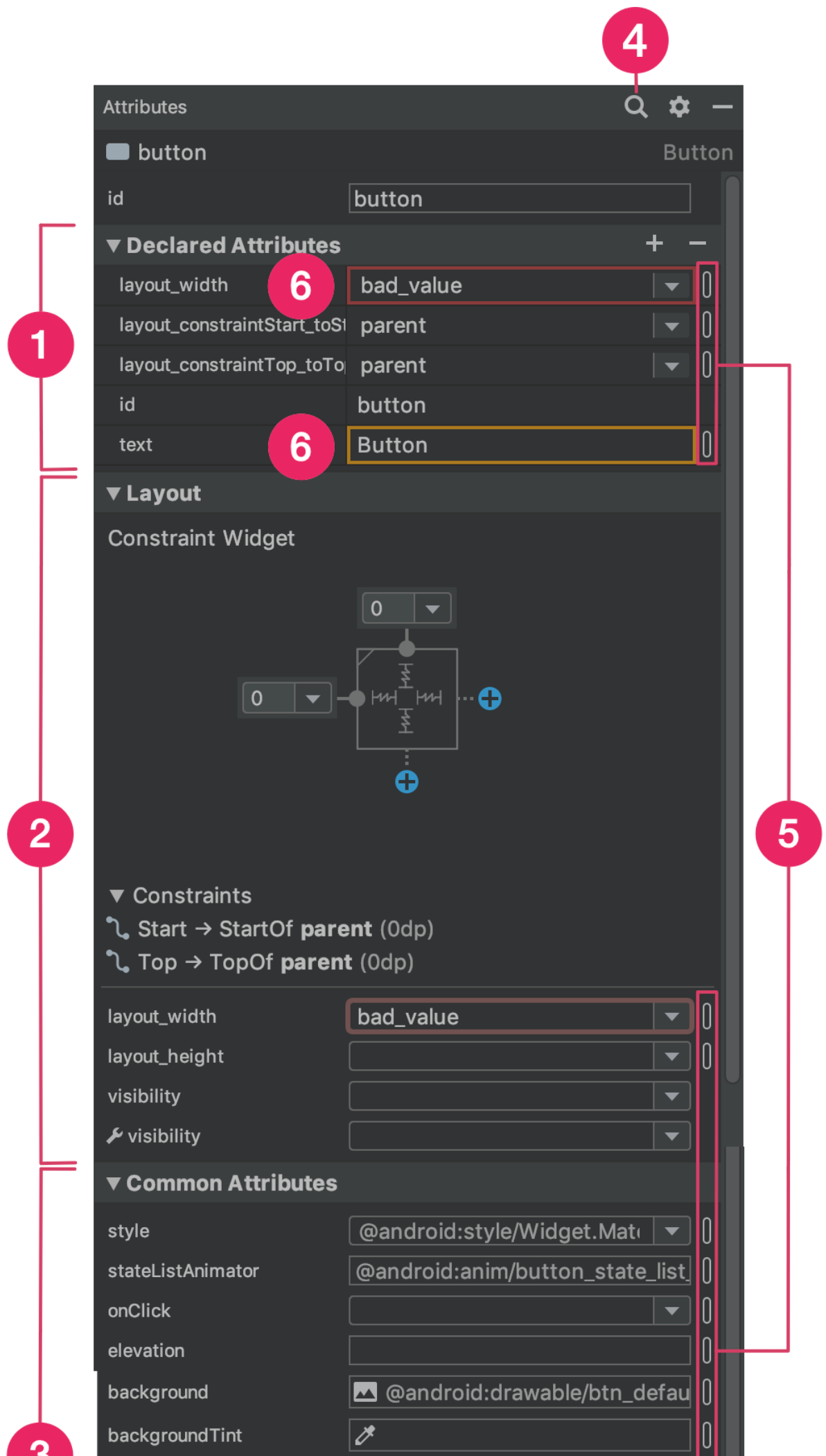
要打开视图或视图组的 Material Guidelines 文档，请右键点击 **Palette** 中的界面元素，然后从上下文菜单中选择 **Material Guidelines**。对于相应内容，如果不存在任何特定条目，则此命令会打开[Material Guidelines 文档](#)的首页。

将视图添加到布局中

要开始构建布局，只需将视图和视图组从 **Palette** 拖动到设计编辑器中即可。将视图放置到布局中后，编辑器会显示有关该视图与布局其余部分的关系的信息。

如果您使用的是 `ConstraintLayout`，则可以使用 Infer Constraints 和 Autoconnect 功能[自动创建约束条件](#)。

修改视图属性



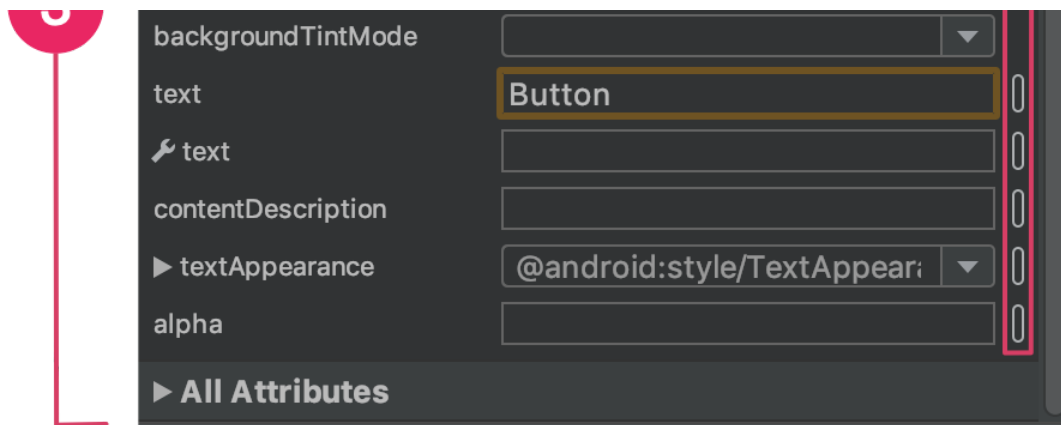





图 3. Attributes 窗口

您可以在 **Attributes** 窗口（位于 Layout Editor 的右侧）中修改视图属性，而不用在 XML 中执行此操作。只有在设计编辑器处于打开状态时，此窗口才会显示，因此请确保您已选择窗口底部的 **Design** 标签页。

当您选择视图时（无论是通过在 **Component Tree** 中点击该视图，还是在设计编辑器中点击该视图），**Attributes** 窗口会显示以下内容，如图 3 所示：

1. **Declared Attributes** 部分，其中会列出布局文件中指定的属性。要添加属性，请点击该部分右上角的 **Add** 图标 .
2. 包含宽度/高度样式控件的视图检查器。对于 `ConstraintLayout` 中的视图，该部分还会显示约束偏差，并列出相应视图使用的约束条件。如需了解详情，请参阅[使用 ConstraintLayout 构建自适应界面](#)。
3. 所选视图的常见属性列表。要查看所有可用的属性，请展开窗口底部的 **All Attributes** 部分。
4. “Search”按钮。点击此按钮可搜索特定的视图属性。
5. 每个属性值右侧都会显示指示器。当属性值是资源引用时，指示器是非中空的 ，否则是空的 。这可让您一目了然地识别硬编码值。点击该指示器（无论其处于哪种状态）都会打开 **Resources** 对话框窗口，您可以在其中选择相应属性的资源引用。
6. 系统会突出显示带有错误或警告的属性，以红色突出显示表示错误，以橙色突出显示表示警告。例如，如果布局定义属性中存在无效条目，系统会显示错误（如图所示）。如果在应该使用资源引用时使用了硬编码值，系统会显示警告（如图所示）。

#### 向视图中添加示例数据

由于很多 Android 布局都依赖于运行时数据，因此在设计应用时很难直观呈现布局的外观和风格。在 Android Studio 3.2 及更高版本中，您可以从布局编辑器中向 `TextView`、`ImageView` 或 `RecyclerView` 添加示例预览数据。

**注意：**当您向 `view` 添加示例数据时，Android Studio 会更改您的项目，就像您在使用自己的数据一样。您随后可以根据需要修改这些更改内容。

您可以右键点击其中某种视图类型，然后选择 **Set Sample Data** 以显示 **Design-time View Attributes** 窗口，如图 4 所示。

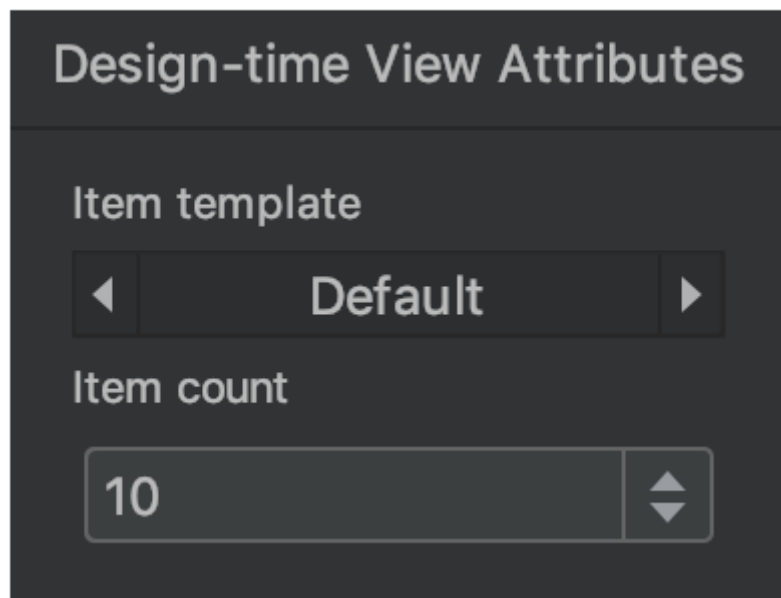


图 4. Design-time View Attributes 窗口

在 `TextView` 中，您可以在不同的示例文本类别之间选择。在使用示例文本时，Android Studio 会使用您选择的示例数据填充 `TextView` 的 `text` 属性。请注意，仅当 `text` 属性为空时，您才能通过 **Design-time View Attributes** 窗口选择示例文本。

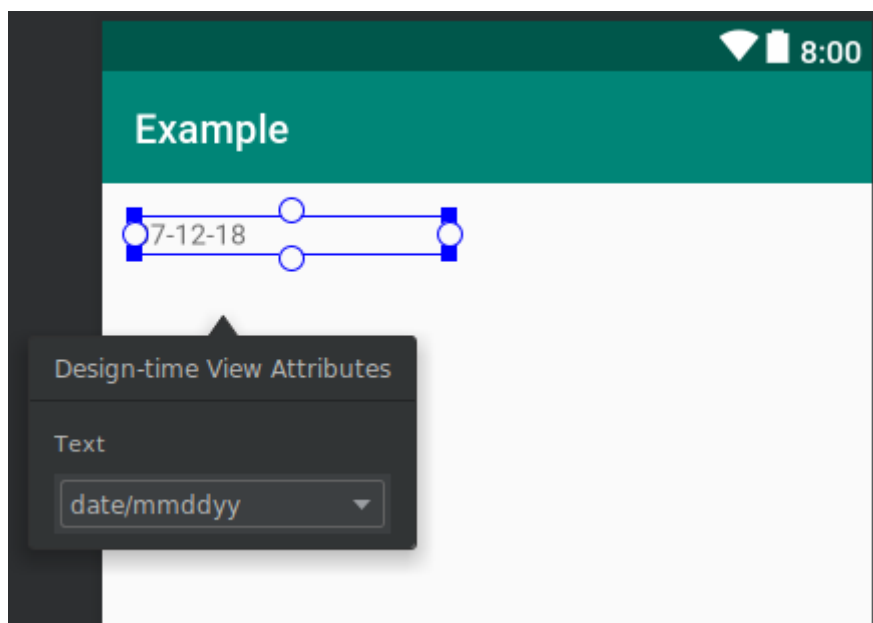
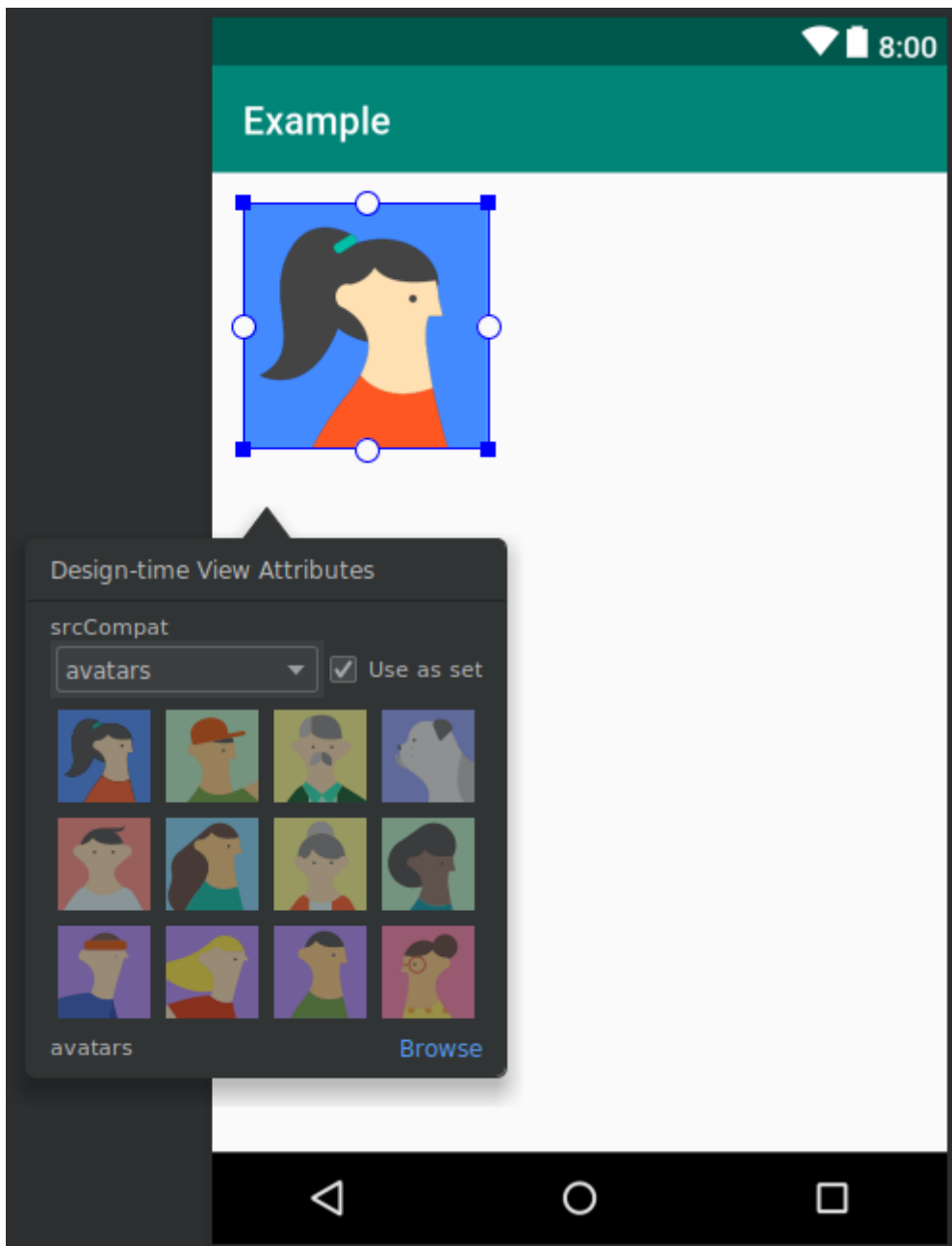


Figure 5. 包含示例数据的 `TextView`

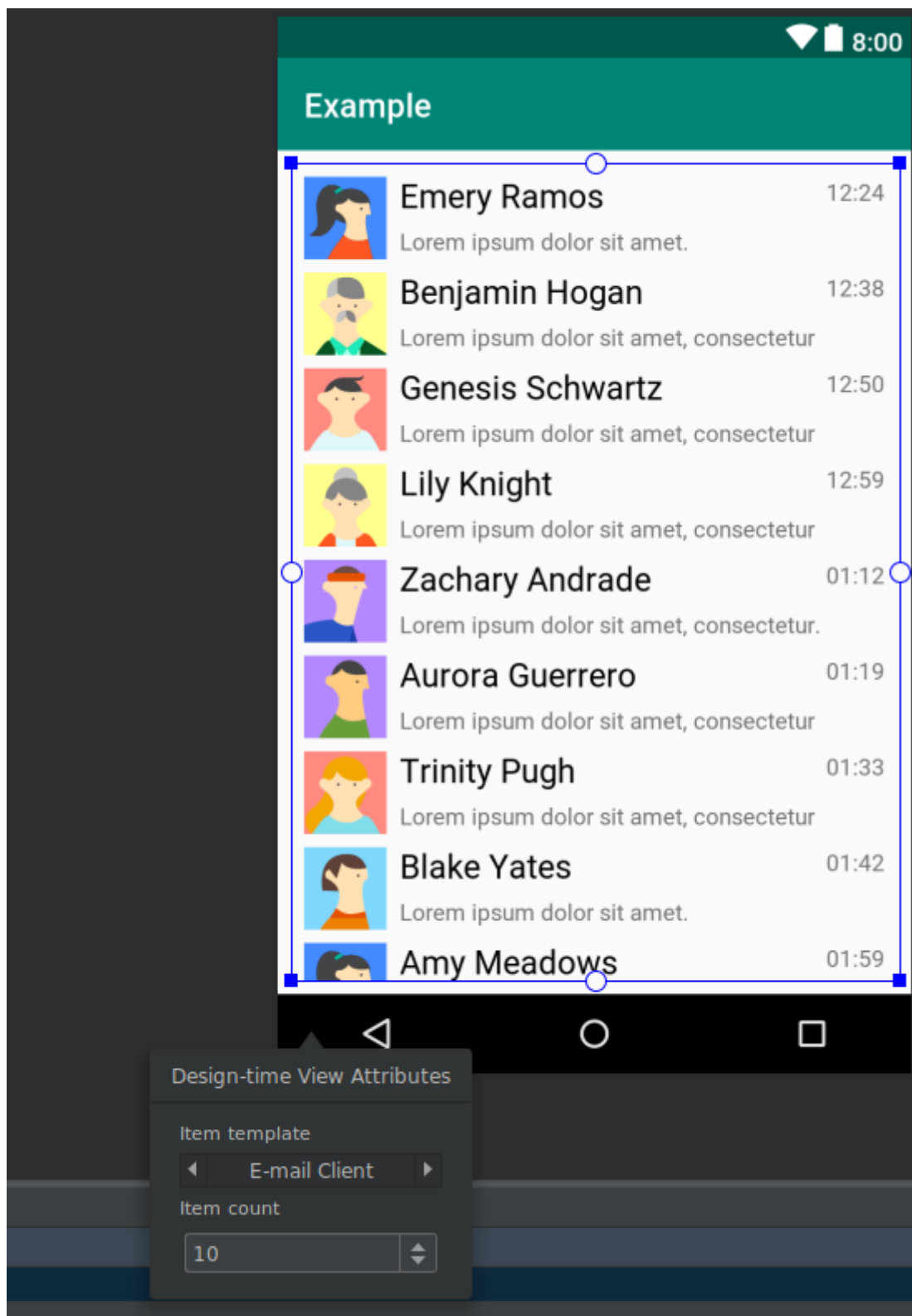
在 `ImageView` 中，您可以在不同的示例图片之间进行选择。当您选择示例图片后，Android Studio 会填充 `ImageView` 的 `tools:src` 属性（或 `tools:srcCompat`，如果使用支持库）。



**Figure 6.** 包含示例数据的 `ImageView`



在 `RecyclerView` 中，您可以在一组包含示例图片和文本的模板之间进行选择。使用这些模板时，Android Studio 会将 `recycler_view_item.xml` 文件添加到 `res/layout` 目录，该文件中包含示例数据的布局。Android Studio 还会将元数据添加到 `RecyclerView`，以正确显示示例数据。







**Figure 7.** 包含示例数据的 `RecyclerView`

显示布局警告和错误

如果在布局中检测到任何问题，系统会在 **Component Tree** 中相应视图的旁边显示感叹号图标（或），以指明存在问题。要查看错误详细信息，请点击该图标。

要在编辑器下方的窗口中查看所有已知问题，请点击工具栏中的 **Show Warnings and Errors**（或）。

在此窗口中，您还可以启用 **Show issues on the preview**，以便向预览中的每个相应视图（仅在 Design 视图中，而非 Blueprint 视图中）添加警告或错误图标。

下载字体并将其应用于文本

使用 Android 8.0（API 级别 26）或 Android 支持库 26.0.0 或更高版本时，您可以按照以下步骤从数百种字体中进行选择：

1. 在 Layout Editor 中，点击 **Design** 标签以在设计编辑器中查看布局。
2. 点击文本视图。
3. 在 **Attributes** 窗口中，展开 **textAppearance**，然后点击以展开 **fontFamily** 方框。
4. 滚动到列表底部，然后点击 **More Fonts**，以打开 **Resources** 对话框。
5. 在 **Resources** 对话框中，通过浏览列表或在顶部的搜索栏中输入字体来选择字体。如果您选择 **Downloadable** 下列出的某种字体，则可以点击 **Create downloadable font** 以在运行时加载该字体（作为[可下载字体](#)），或点击 **Add font to project** 以将 TTF 字体文件打包到 APK 中。  
（Android 系统提供了 **Android** 下列出的字体，因此这些字体不需要下载，也不需要打包到 APK 中。）
6. 点击 **OK**。